CUNY SPS DATA 621 - CTG5 - HW3

Betsy Rosalen, Gabrielle Bartomeo, Jeremy O'Brien, Lidiia Tronina, Rose Koh

April 10th, 2019

Contents

1 DATA EXPLORATION

Relocating to a new city or state can be very stressful. In addition to the stress of packing and moving, you may also be nervous about moving to an unfamiliar area. To better understand their new community, some new residents or people interested in moving to a new city choose to review crime statistics in and around their neighborhood. Crime rate may also influence where people choose to live, raise their families and run their businesses; many potential new residents steer clear of cities with higher than average crime rates.

Data was collected in order to predict whether the neighborhood will be at risk for high crime levels. For each neighborhood the response variable, `target`, represents whetever the crime rate is above the median crime rate or not. In addition to that 13 predictor variables were collected representing each neighborhood's: proportion of large lots, non-retail business acres, whether or not it borders the Charles River, nitrogen oxides concentration, average number of rooms per dwelling, proportion of owner-occupied units, distances to five Boston employment centers, accessibility to radial highways, property tax rate, pupil-teacher ratio, proportion of african Americans, percent lower status, and median value of homes. The evaluation data contains the same 13 predictor variables and no target variable so it will be impossible to check the accuracy of our predictions from the testing data.

VARIABLE NAME	DEFINITION	TYPE
target	whether the crime rate is above the median crime rate (1) or not (0)	response variable
zn	proportion of residential land zoned for large lots (over 25000 square feet)	predictor variable
indus	proportion of non-retail business acres per suburb	predictor variable
chas	a dummy var. for whether the suburb borders the Charles River (1) or not (0)	predictor variable
nox	nitrogen oxides concentration (parts per 10 million)	predictor variable
rm	average number of rooms per dwelling	predictor variable
age	proportion of owner-occupied units built prior to 1940	predictor variable
dis	weighted mean of distances to five Boston employment centers	predictor variable
rad	index of accessibility to radial highways	predictor variable
tax	full-value property-tax rate per \$10,000	predictor variable
ptratio	pupil-teacher ratio by town	predictor variable
black	$1000(B_k - 0.63)^2$ where B_k is the proportion of blacks by town	predictor variable
lstat	lower status of the population (percent)	predictor variable
medv	median value of owner-occupied homes in \$1000s	predictor variable

1.1 Summary Statistics

Looking at the Table 2, we can see that `chas` and `target` are binary variables. 49% of our target variable is coded as 0's indicating that the crime rate is NOT above the median crime rate. There are potential outliers present in `zn`, `lstat`, `medv` and `dis`.

1.2 Shape of Predictor Distributions

Figure. 1 shows that the distribution of most of the variables seems skewed. There are some outliers in the right tail of `tax` , `rad`, `medv`, `lstat`, `dis` and left tail of `ptratio`.

Table 2: Summary statistics

	n	min	mean	median	max	sd
zn	466	0.0000	11.5772532	0.00000	100.0000	23.3646511
indus	466	0.4600	11.1050215	9.69000	27.7400	6.8458549
chas	466	0.0000	0.0708155	0.00000	1.0000	0.2567920
nox	466	0.3890	0.5543105	0.53800	0.8710	0.1166667
rm	466	3.8630	6.2906738	6.21000	8.7800	0.7048513
age	466	2.9000	68.3675966	77.15000	100.0000	28.3213784
dis	466	1.1296	3.7956929	3.19095	12.1265	2.1069496
rad	466	1.0000	9.5300429	5.00000	24.0000	8.6859272
tax	466	187.0000	409.5021459	334.50000	711.0000	167.9000887
ptratio	466	12.6000	18.3984979	18.90000	22.0000	2.1968447
lstat	466	1.7300	12.6314592	11.35000	37.9700	7.1018907
medv	466	5.0000	22.5892704	21.20000	50.0000	9.2396814
target	466	0.0000	0.4914163	0.00000	1.0000	0.5004636

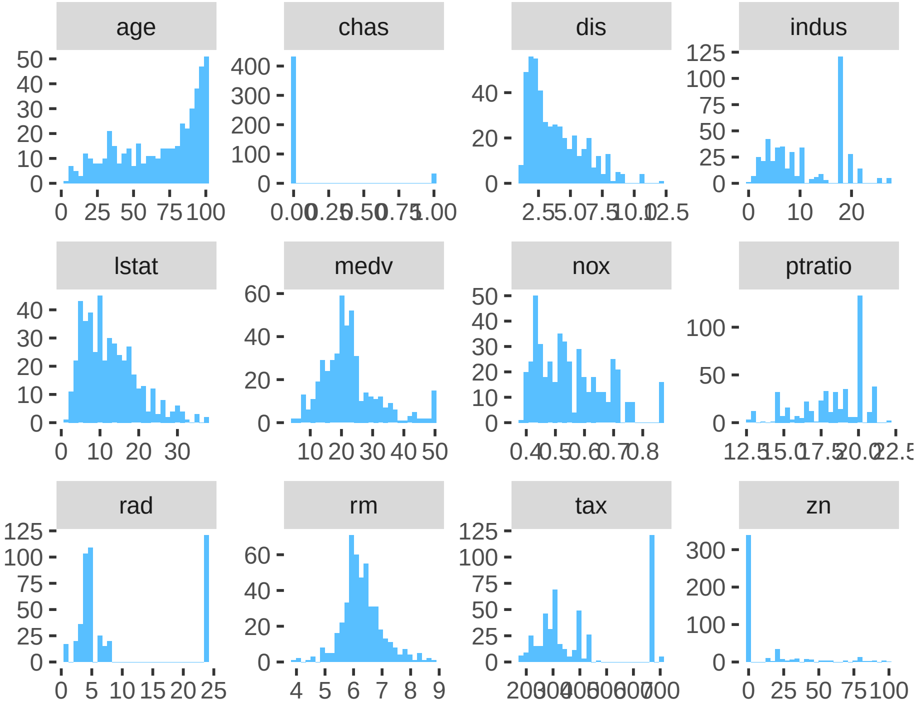

Figure 1: Data Distributions

1.3 Outliers

Figure. 2 shows that there are also a large number of outliers that need to be accounted for, most significantly in `zn` and `medv` and less significantly in `lstat`, `dis` and `rm`. Since `tax` variable has values which are very large compared to other variables in the dataset, it was scaled to fit the boxplot by dividing by 10.

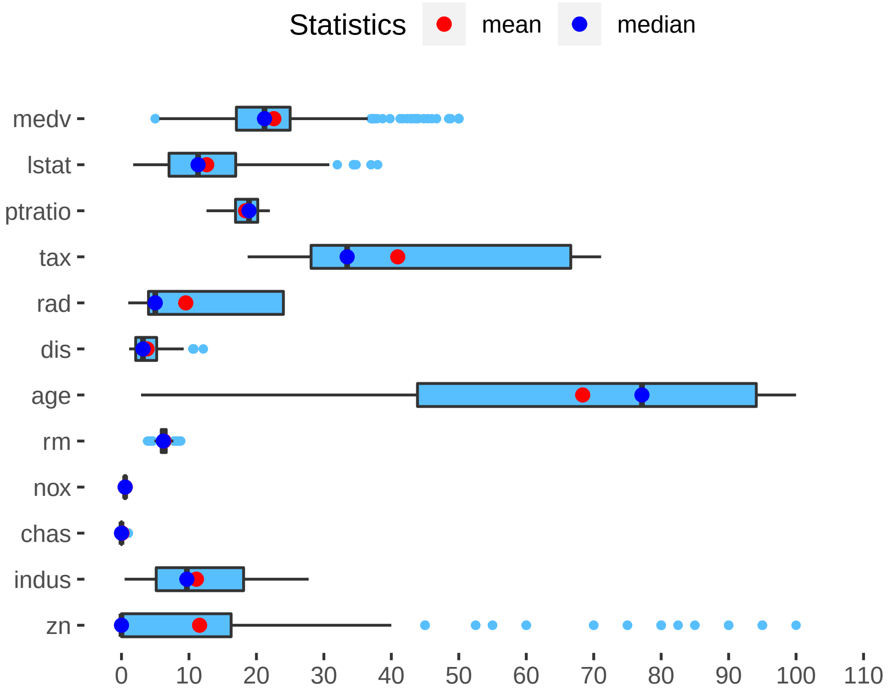

Figure 2: Boxplots highlighting many outliers in the data.

1.4 Missing Values

There are no missing values in any of our observations gathered across the thirteen predictor variables as can be seen in Figure. 3.

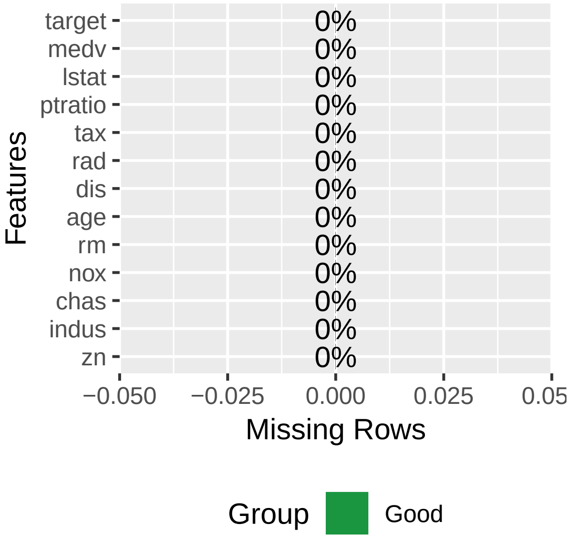

Figure 3: Missing values

1.5 Linearity

Each variable was plotted against the target variable in order to determine at a glance which had the most potential linearity before the dataset was modified.

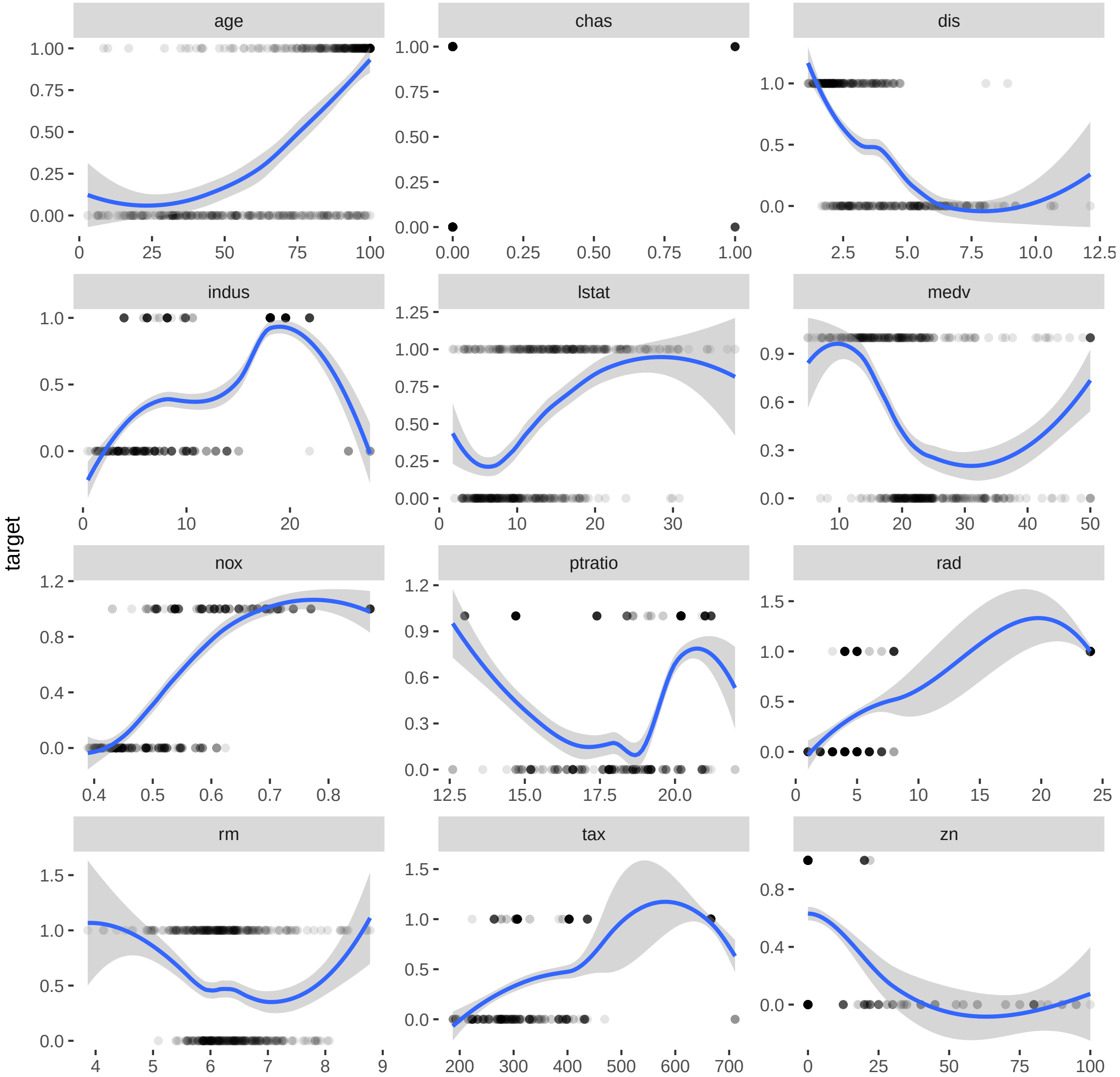

Figure 4: Linear relationships between each predictors and the target

As can be observed in Figure. 4, the most influential variables are the ones previously discussed to have severe outliers and skew, and their linear relationship is negative - the higher the variable, the lower the target wins.

2 DATA PREPARATION

2.1 Missing Values and NA Imputation

Given that the training dataset does include missing values, there's not need to make systematic corrections or imputations.

2.2 Dealing with outliers, leverage, and influence points

While logistic regression can be more robust to leverage points (explantory variable values, which are distant on the x-axis), outliers (response variable values, which are distant on the y-axis) can exert influence which affects the curve and accuracy of target predictions.

- `dis`, `tax` (property tax rate per $10k), and `medv` (median value of owner-occupied homes) see a few outliers and leverage points in both target classes
- `indus` (the non-retail business acreage proportion) and `lstat` (percent lower status population) both have outliers in the below-mean (0) class
- `ptratio` (pupil-teacher ratio) fit is very impacted by density of low values in the above-mean class, making the linear relationship appear parabolic
- `rad` (highway access index) is influenced by a high-value concentration of locations distant from radial highways that fall in the above-mean class
- `rm` (average rooms per dwelling) sees a wider distribution of house size for tha above-mean class then the below-mean; while `zn` (large-lot zoned land proportion) sees the opposite, with a concentration around a few non-residential land proportions for the above-mean class and a wide dispersion for the below-mean class

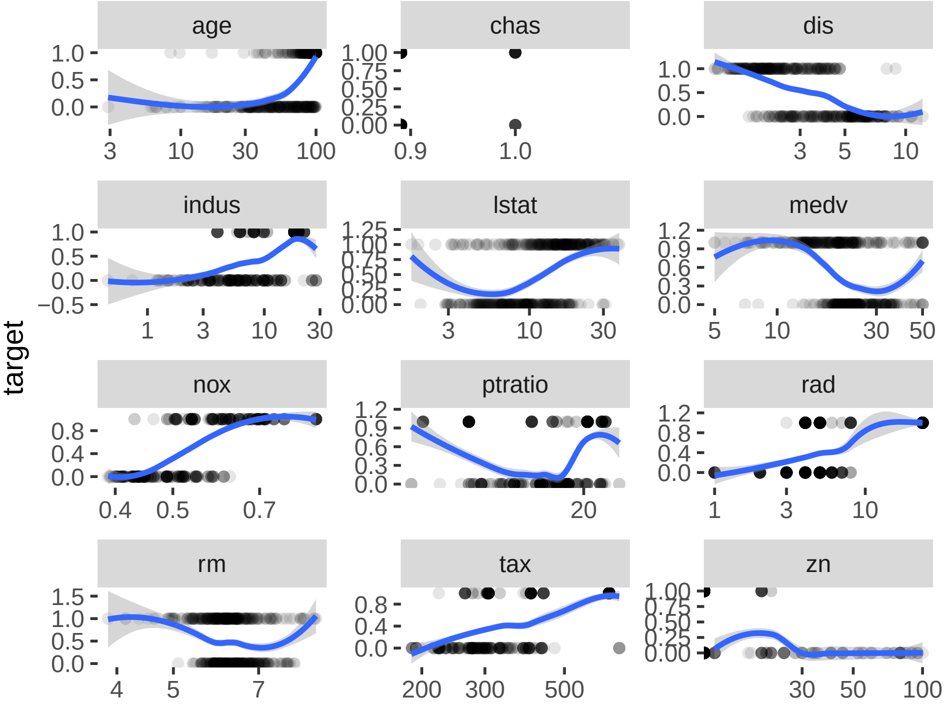

Figure 5: Linear relationships between each predictors and the target

We examined the linear relationships after a log transformation, which smoothed several relationships but still demonstrated visible influence for several variables: `lstat`, `medv`, `ptratio`, `rad`, `rm`, `tax`, and `zn`. We discuss further in the feature engineering section below.

[JEREMY: TEAM, SO WE WANT TO DO FURTHER INVESTIGATION OF OUTLIERS, LOOKING AT R2, STANDARD ERRORS, P-VALS, AND LEVERAGE VALUES FROM THE HAT MATRIX FOR PAIRS OF MODELS, ONE THAT INCLUDES OUTLIERS AND ANOTHER THAT DOESN'T; OR IS THIS ENOUGH?]

2.3 Correlation

```
##                  zn       indus        chas         nox          rm
## zn       1.00000000 -0.53826643 -0.04016203 -0.51704518  0.31981410
## indus   -0.53826643  1.00000000  0.06118317  0.75963008 -0.39271181
## chas    -0.04016203  0.06118317  1.00000000  0.09745577  0.09050979
## nox     -0.51704518  0.75963008  0.09745577  1.00000000 -0.29548972
## rm       0.31981410 -0.39271181  0.09050979 -0.29548972  1.00000000
## age     -0.57258054  0.63958182  0.07888366  0.73512782 -0.23281251
## dis      0.66012434 -0.70361886 -0.09657711 -0.76888404  0.19901584
## rad     -0.31548119  0.60062839 -0.01590037  0.59582984 -0.20844570
## tax     -0.31928408  0.73222922 -0.04676476  0.65387804 -0.29693430
## ptratio -0.39103573  0.39468980 -0.12866058  0.17626871 -0.36034706
## lstat   -0.43299252  0.60711023 -0.05142322  0.59624264 -0.63202445
## medv     0.37671713 -0.49617432  0.16156528 -0.43012267  0.70533679
##                 age         dis         rad         tax    ptratio
## zn      -0.57258054  0.66012434 -0.31548119 -0.31928408 -0.3910357
## indus    0.63958182 -0.70361886  0.60062839  0.73222922  0.3946898
## chas     0.07888366 -0.09657711 -0.01590037 -0.04676476 -0.1286606
## nox      0.73512782 -0.76888404  0.59582984  0.65387804  0.1762687
## rm      -0.23281251  0.19901584 -0.20844570 -0.29693430 -0.3603471
## age      1.00000000 -0.75089759  0.46031430  0.51212452  0.2554479
## dis     -0.75089759  1.00000000 -0.49499193 -0.53425464 -0.2333394
## rad      0.46031430 -0.49499193  1.00000000  0.90646323  0.4714516
## tax      0.51212452 -0.53425464  0.90646323  1.00000000  0.4744223
## ptratio  0.25544785 -0.23333940  0.47145160  0.47442229  1.0000000
## lstat    0.60562001 -0.50752800  0.50310125  0.56418864  0.3773560
## medv    -0.37815605  0.25669476 -0.39766826 -0.49003287 -0.5159153
##               lstat       medv
## zn      -0.43299252  0.3767171
## indus    0.60711023 -0.4961743
## chas    -0.05142322  0.1615653
## nox      0.59624264 -0.4301227
## rm      -0.63202445  0.7053368
## age      0.60562001 -0.3781560
## dis     -0.50752800  0.2566948
## rad      0.50310125 -0.3976683
## tax      0.56418864 -0.4900329
## ptratio  0.37735605 -0.5159153
## lstat    1.00000000 -0.7358008
## medv    -0.73580078  1.0000000
```

An examination of correlation between the explanatory variables reveals the following:

- `indus` (non-retail business acre proportion) is positively correlated with `nox` (pollution concentration, $r = .76$) and `tax` (property tax rate per \$10k, $r = .73$) and is negatively correlated with `dis` (weighted mean distance to employment centers, $r = -.7$)

- `chas` (bordering Charles river) correlated with `nox` ($r = .97$) and `rm` (average rooms per dwelling, $r = .91$) and `age` (proportion of pre-1940 homes, $r = .79$); and is negatively correlated with `dis` ($r = -.97$)
- `medv` (median value of owner-occupied homes) is correlated with `rm` ($r = .71$); and is negatively correlated with `lstat` (percent lower status population, $r = -.74$)
- `age` is correlated with `nox` ($r = .74$); and is negatively correlated with `dis` ($r = -.75$)
- `rad` (highway access index) correlated with `tax` ($r = .91$)

[JEREMY: TEAM, LET'S DISCUSS HOW WE'D LIKE TO APPLY THESE CORRELATION FINDINGS TO MODEL EVALUATION AND VARIABLE SELECTION]

2.4 Feature Engineering

In MARR, Sheather quotes Cook and Weisberg, suggesting that the best way to determine need for log transformation of skewed predictors is to include both the original and transformed varaibles in the logistic regression model in order assess their relative contributions directly and prune accordingly

Reexamining the histograms of the predictor distributions above reveals that:

- `age` is left-skewed
- `dis` is right-skewed, and `zn` is extremely so
- `nox` is right-skewed and platykurtic (thin-tailed)
- `rad` and `tax` seem to have normal distributions, with large numbers of outliers at particular levels
- `indus` and `ptratio` reveal pecular skew, with incidences at particular high level, perhaps due to regulation or infrastructure requirements

We include log transforms of `age`, `dis`, `nox`, `rad`, `tax`, `indus`, and `ptratio` in the dataset for evaluation in models.

[JEREMY: TEAM, DO WE WANT TO ADDRESS THIS BY CALLING log() ON VARIABLES WHEN BUILDING glm(), OR SHOULD WE TRANSFORM IN SOURCE DATASET? I'VE ASSUMED THE FORMER.]

3 BUILD MODELS

Due to small number of observations for training, the K-fold cross validation is used to train with k=10. We will split the data and hold 20% for validation for modelling. When we the final model is selected, the model can be applied to the full training set.

3.1 Model 1

$\widehat{y} = -1.95\times zn - 0.44\times indus + 0.12\times chas + 5.65\times nox - 0.06\times rm + 0.71\times age + 1.47\times dis + 5.54\times rad - 0.85\times tax + 0.66\times ptratio + 0.$

The First model is the binary logistic model including all the explanatory variables. The data is centered and scaled based on the mean and standard deviation of the variables.

```
##
## Call:
## NULL
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.93085  -0.09660  -0.00017   0.00092   2.62158
##
```

```
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.56944    1.02719   2.501 0.012369 *
## zn          -2.83263    1.28439  -2.205 0.027424 *
## indus       -0.33038    0.42008  -0.786 0.431586
## chas         0.03407    0.23433   0.145 0.884410
## nox          6.36083    1.22651   5.186 2.15e-07 ***
## rm          -0.25938    0.63754  -0.407 0.684127
## age          0.90829    0.49524   1.834 0.066646 .
## dis          1.56066    0.62778   2.486 0.012919 *
## rad          7.01175    1.92828   3.636 0.000277 ***
## tax         -1.25785    0.62323  -2.018 0.043561 *
## ptratio      0.65062    0.35166   1.850 0.064297 .
## lstat        0.48253    0.48462   0.996 0.319407
## medv         1.90667    0.86442   2.206 0.027402 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 516.76  on 372  degrees of freedom
## Residual deviance: 130.45  on 360  degrees of freedom
## AIC: 156.45
##
## Number of Fisher Scoring iterations: 9
```

The residual deviance is 147.10 and the AIC is 173.1. We will consider this as the baseline for all models.

	x
zn	613.67283
indus	65.64561
chas	20.42626
nox	559.60619
rm	151.20441
age	91.23600
dis	146.60920
rad	1383.19099
tax	144.48897
ptratio	46.00402
lstat	87.36810
medv	277.96351

The review of the VIF output suggests that some variables are highly colinear and may not be necessary to build a model.

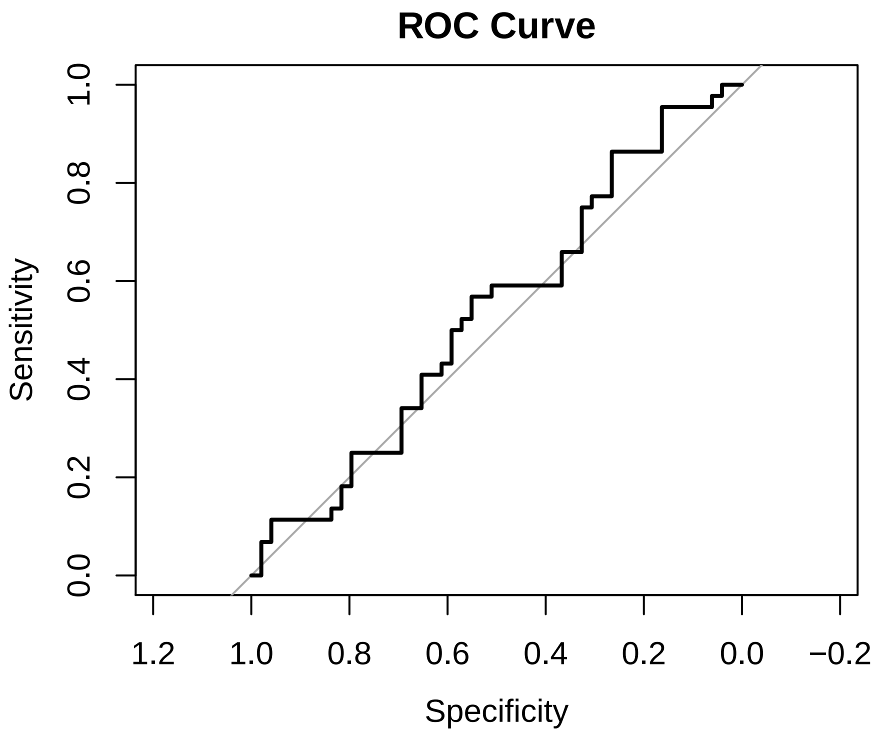

Figure 6: Model 1 ROC Curve

```
## Area under the curve: 0.5353
```

3.2 Model 2

The logarithmic transformation on explanatory variables is used for Model 2 in order to normalize the distribution of the explanatory variables. Model 2 also removes variables that seemed unnecessary in Model 1.

```
##
## Call:
## NULL
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.92902  -0.23761  -0.00438   0.13580   3.04723
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.2809     0.3066   0.916  0.35965
## zn           -0.6950     0.5667  -1.226  0.22002
## log_nox       4.2975     0.7370   5.831 5.50e-09 ***
## log_age       0.5885     0.3542   1.662  0.09660 .
## log_dis       1.2955     0.4276   3.030  0.00245 **
## log_rad       2.0024     0.4438   4.512 6.43e-06 ***
## log_ptratio   0.6588     0.2678   2.460  0.01389 *
## log_medv      1.0991     0.3468   3.169  0.00153 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 516.87  on 372  degrees of freedom
## Residual deviance: 186.49  on 365  degrees of freedom
## AIC: 202.49
##
## Number of Fisher Scoring iterations: 7
```

	x
zn	119.46449
log_nox	202.04310
log_age	46.65750
log_dis	68.02166
log_rad	73.27089
log_ptratio	26.67770
log_medv	44.74688

Unfortunately, both Residual deviance and AIC have worsened and the log transformation may not have been the optimal choice.

3.3 Model 3

Model 3 removes the variables with high VIF values from Model 2.

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7947  -0.4275  -0.0108   0.1957   3.3763
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.08758    0.24995   0.350 0.726043
## log_dis     -0.82775    0.25257  -3.277 0.001048 **
## log_age      1.48228    0.40333   3.675 0.000238 ***
## log_rad      2.18101    0.39063   5.583 2.36e-08 ***
## log_ptratio -0.02881    0.17762  -0.162 0.871145
## log_medv     0.06645    0.23837   0.279 0.780417
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 517.06  on 372  degrees of freedom
## Residual deviance: 225.38  on 367  degrees of freedom
## AIC: 237.38
##
## Number of Fisher Scoring iterations: 7
```

	x
log_dis	23.73105
log_age	60.51441
log_rad	56.76553
log_ptratio	11.73625
log_medv	21.13631

[Since model 2, 3 is not developing, I thought I should try the step approach...]

Before we proceed to the next model, Consider different perspective on the built models. Use step approach to see if it is possible to build a model iteratively (forward/backward) and check if there is a room to improve.

```
## # weights:  14 (13 variable)
## initial  value 323.006586
## iter  10 value 161.593045
## iter  20 value 102.106425
## iter  30 value 96.789997
## iter  40 value 96.053174
## iter  50 value 96.023666
## final  value 96.023454
## converged
## Start:  AIC=218.05
## target ~ zn + indus + chas + nox + rm + age + dis + rad + tax +
##     ptratio + lstat + medv
##
## trying - zn
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 162.680398
## iter  20 value 99.503346
## iter  30 value 98.540288
## iter  40 value 98.446227
## final  value 98.445983
## converged
## trying - indus
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 160.810013
## iter  20 value 102.695797
## iter  30 value 98.155939
## iter  40 value 97.005071
## iter  50 value 96.992784
## iter  50 value 96.992784
## iter  50 value 96.992784
## final  value 96.992784
## converged
## trying - chas
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 161.612881
## iter  20 value 102.700944
## iter  30 value 97.729620
## iter  40 value 96.788707
## final  value 96.762343
## converged
## trying - nox
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 161.774353
## iter  20 value 135.075737
## iter  30 value 132.524852
## iter  40 value 132.522871
## final  value 132.522862
## converged
## trying - rm
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 166.648639
## iter  20 value 102.184025
## iter  30 value 97.102636
## iter  40 value 96.368857
## final  value 96.355390
## converged
## trying - age
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 174.917453
## iter  20 value 103.983372
## iter  30 value 99.715274
## iter  40 value 99.376038
## final  value 99.362783
## converged
## trying - dis
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 162.436164
## iter  20 value 106.159378
## iter  30 value 102.470242
## iter  40 value 102.003310
## final  value 101.920611
## converged
## trying - rad
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 190.999955
## iter  20 value 119.086567
## iter  30 value 116.873349
## iter  40 value 116.871968
## final  value 116.871842
## converged
## trying - tax
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 154.765721
## iter  20 value 98.361892
## iter  30 value 98.294745
## iter  40 value 98.293355
## iter  40 value 98.293354
## iter  40 value 98.293354
## final  value 98.293354
## converged
## trying - ptratio
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 164.408271
## iter  20 value 103.789174
## iter  30 value 101.909757
## iter  40 value 101.683202
## final  value 101.662191
## converged
## trying - lstat
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 148.629217
## iter  20 value 99.060598
## iter  30 value 96.982364
## iter  40 value 96.384387
## final  value 96.383587
## converged
## trying - medv
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 153.754301
## iter  20 value 102.865693
## iter  30 value 100.540227
## iter  40 value 99.990701
## final  value 99.975286
## converged
##           Df      AIC
## - rm      12 216.7108
## - lstat   12 216.7672
## - chas    12 217.5247
## - indus   12 217.9856
## <none>    13 218.0469
## - tax     12 220.5867
## - zn      12 220.8920
## - age     12 222.7256
## - medv    12 223.9506
## - ptratio 12 227.3244
## - dis     12 227.8412
## - rad     12 257.7437
## - nox     12 289.0457
## # weights:  13 (12 variable)
## initial  value 323.006586
## iter  10 value 166.648639
## iter  20 value 102.184025
## iter  30 value 97.102636
## iter  40 value 96.368857
## final  value 96.355390
## converged
##
## Step:  AIC=216.71
## target ~ zn + indus + chas + nox + age + dis + rad + tax + ptratio +
##     lstat + medv
##
## trying - zn
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 150.869421
## iter  20 value 99.201156
## iter  30 value 99.053630
## iter  40 value 99.035050
## final  value 99.034097
## converged
## trying - indus
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 148.208413
## iter  20 value 99.498413
## iter  30 value 97.891871
## iter  40 value 97.302491
## final  value 97.290928
## converged
## trying - chas
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 166.729107
## iter  20 value 102.084717
## iter  30 value 97.748690
## iter  40 value 97.141061
## final  value 97.122056
## converged
## trying - nox
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 166.912140
## iter  20 value 134.814016
## iter  30 value 132.576661
## final  value 132.569005
## converged
## trying - age
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 155.195445
## iter  20 value 102.166861
## iter  30 value 99.856998
## iter  40 value 99.569239
## final  value 99.555222
## converged
## trying - dis
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 159.643756
## iter  20 value 104.173732
## iter  30 value 102.576993
## iter  40 value 101.983097
## final  value 101.923692
## converged
## trying - rad
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 176.803416
## iter  20 value 117.399576
## iter  30 value 116.918782
## iter  40 value 116.907385
## final  value 116.907339
## converged
## trying - tax
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 147.767790
## iter  20 value 98.908209
## iter  30 value 98.802685
## final  value 98.792520
## converged
## trying - ptratio
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 152.688860
## iter  20 value 103.289787
## iter  30 value 101.889396
## iter  40 value 101.779104
## final  value 101.767584
## converged
## trying - lstat
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 148.750702
## iter  20 value 100.346123
## iter  30 value 97.685184
## iter  40 value 97.176320
## final  value 97.162394
## converged
## trying - medv
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 143.945237
## iter  20 value 104.526080
## iter  30 value 103.450311
## iter  40 value 102.677018
## final  value 102.676980
## converged
##           Df      AIC
## - chas    11 216.2441
## - lstat   11 216.3248
## - indus   11 216.5819
## <none>    12 216.7108
## - tax     11 219.5850
## - zn      11 220.0682
## - age     11 221.1104
## - ptratio 11 225.5352
## - dis     11 225.8474
## - medv    11 227.3540
## - rad     11 255.8147
## - nox     11 287.1380
## # weights:  12 (11 variable)
## initial  value 323.006586
## iter  10 value 166.729107
## iter  20 value 102.084717
## iter  30 value 97.748690
## iter  40 value 97.141061
## final  value 97.122056
## converged
##
## Step:  AIC=216.24
## target ~ zn + indus + nox + age + dis + rad + tax + ptratio +
##     lstat + medv
##
## trying - zn
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 146.071503
## iter  20 value 100.527231
## iter  30 value 100.368794
## iter  40 value 100.297378
## final  value 100.296646
## converged
## trying - indus
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 147.981203
## iter  20 value 100.223103
## iter  30 value 98.439825
## iter  40 value 97.757538
## final  value 97.757342
## converged
## trying - nox
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 166.989867
## iter  20 value 134.092128
## iter  30 value 132.823390
## final  value 132.593314
## converged
## trying - age
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 155.190453
## iter  20 value 103.529068
## iter  30 value 100.889271
## iter  40 value 100.507275
## final  value 100.501623
## converged
## trying - dis
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 158.846947
## iter  20 value 104.752040
## iter  30 value 102.895749
## iter  40 value 102.420467
## final  value 102.415440
## converged
## trying - rad
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 176.185036
## iter  20 value 120.925604
## iter  30 value 120.732096
## iter  40 value 120.706361
## final  value 120.706356
## converged
## trying - tax
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 145.665552
## iter  20 value 100.447423
## iter  30 value 100.378536
## iter  40 value 100.372763
## final  value 100.372635
## converged
## trying - ptratio
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 152.687951
## iter  20 value 103.248800
## iter  30 value 102.323054
## iter  40 value 101.971936
## final  value 101.971283
## converged
## trying - lstat
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 148.348909
## iter  20 value 100.778692
## iter  30 value 98.498157
## iter  40 value 98.165477
## final  value 98.164052
## converged
## trying - medv
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 143.702386
## iter  20 value 105.586465
## iter  30 value 104.438834
## iter  40 value 103.561470
## final  value 103.561150
## converged
##           Df      AIC
## - indus   10 215.5147
## <none>    11 216.2441
## - lstat   10 216.3281
## - zn      10 220.5933
## - tax     10 220.7453
## - age     10 221.0032
## - ptratio 10 223.9426
## - dis     10 224.8309
## - medv    10 227.1223
## - rad     10 261.4127
## - nox     10 285.1866
## # weights:  11 (10 variable)
## initial  value 323.006586
## iter  10 value 147.981203
## iter  20 value 100.223103
## iter  30 value 98.439825
## iter  40 value 97.757538
## final  value 97.757342
## converged
##
## Step:  AIC=215.51
## target ~ zn + nox + age + dis + rad + tax + ptratio + lstat +
##     medv
##
## trying - zn
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 140.661083
## iter  20 value 101.571692
## iter  30 value 101.028362
## final  value 101.027047
## converged
## trying - nox
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 152.363183
## iter  20 value 135.445980
## iter  30 value 135.311628
## final  value 135.296886
## converged
## trying - age
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 146.253647
## iter  20 value 103.790445
## iter  30 value 101.139192
## final  value 101.116144
## converged
## trying - dis
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 142.845974
## iter  20 value 104.107896
## iter  30 value 103.124802
## final  value 102.980209
## converged
## trying - rad
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 165.547617
## iter  20 value 124.872289
## iter  30 value 124.775479
## final  value 124.774855
## converged
## trying - tax
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 135.974361
## iter  20 value 103.436454
## iter  30 value 103.307167
## final  value 103.301765
## converged
## trying - ptratio
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 142.115672
## iter  20 value 103.674685
## iter  30 value 102.576011
## final  value 102.504123
## converged
## trying - lstat
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 140.433888
## iter  20 value 100.961973
## iter  30 value 98.672610
## final  value 98.661421
## converged
## trying - medv
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 139.134764
## iter  20 value 106.619542
## iter  30 value 104.076301
## final  value 104.067084
## converged
##           Df      AIC
## - lstat    9 215.3228
## <none>    10 215.5147
## - zn       9 220.0541
## - age      9 220.2323
## - ptratio  9 223.0082
## - dis      9 223.9604
## - tax      9 224.6035
## - medv     9 226.1342
## - rad      9 267.5497
## - nox      9 288.5938
## # weights:  10 (9 variable)
## initial  value 323.006586
## iter  10 value 140.433888
## iter  20 value 100.961973
## iter  30 value 98.672610
## final  value 98.661421
## converged
##
## Step:  AIC=215.32
## target ~ zn + nox + age + dis + rad + tax + ptratio + medv
##
## trying - zn
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 127.585845
## iter  20 value 101.954978
## iter  30 value 101.724767
## final  value 101.723670
## converged
## trying - nox
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 150.841925
## iter  20 value 137.723673
## iter  30 value 136.587618
## final  value 136.587609
## converged
## trying - age
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 137.886175
## iter  20 value 105.922859
## iter  30 value 103.618171
## final  value 103.566205
## converged
## trying - dis
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 118.218405
## iter  20 value 105.056849
## iter  30 value 103.830346
## final  value 103.821592
## converged
## trying - rad
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 153.933370
## iter  20 value 125.680348
## iter  30 value 125.499957
## final  value 125.490871
## converged
## trying - tax
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 126.581624
## iter  20 value 104.002129
## iter  30 value 103.811666
## final  value 103.807786
## converged
## trying - ptratio
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 137.559319
## iter  20 value 104.240255
## iter  30 value 103.151993
## final  value 103.135670
## converged
## trying - medv
## # weights:  9 (8 variable)
## initial  value 323.006586
## iter  10 value 122.682253
## iter  20 value 106.681432
## iter  30 value 104.325503
## final  value 104.324234
## converged
##           Df      AIC
## <none>     9 215.3228
## - zn       8 219.4473
## - ptratio  8 222.2713
## - age      8 223.1324
## - tax      8 223.6156
## - dis      8 223.6432
## - medv     8 224.6485
## - rad      8 266.9817
## - nox      8 289.1752
```

```
## Call:
## multinom(formula = target ~ zn + nox + age + dis + rad + tax + 
##     ptratio + medv, data = train)
## 
## Coefficients:
##   (Intercept)            zn           nox           age           dis
## -37.420004457  -0.068657565  42.812279216   0.032954647   0.655005663
##           rad           tax       ptratio          medv
##   0.725172172  -0.007757457   0.323670155   0.110489138
## 
## Residual Deviance: 197.3228 
## AIC: 215.3228
```

```
## # weights:  14 (13 variable)
## initial  value 323.006586 
## iter  10 value 161.593045
## iter  20 value 102.106425
## iter  30 value 96.789997
## iter  40 value 96.053174
## iter  50 value 96.023666
## final  value 96.023454 
## converged
## Start:  AIC=218.05
## target ~ zn + indus + chas + nox + rm + age + dis + rad + tax + 
##     ptratio + lstat + medv
## Call:
## multinom(formula = target ~ zn + indus + chas + nox + rm + age +
##     dis + rad + tax + ptratio + lstat + medv, data = train)
##
## Coefficients:
##   (Intercept)            zn         indus          chas           nox
## -40.822813404  -0.065945741  -0.064613819   0.910758951  49.122184372
##            rm           age           dis           rad           tax
##  -0.587489921   0.034188962   0.738659039   0.666362966  -0.006171376
##       ptratio         lstat          medv
##   0.402564041   0.045868448   0.180823739
##
## Residual Deviance: 192.0469
## AIC: 218.0469
```

As above, judging from both Residual Deviance and AIC, the forward/backward stepwise approach did not improve the model.

3.4 Model 4

```
##
## Call:
## glm(formula = target ~ log_dis + log_age + log_rad, family = binomial(logit),
##     data = split.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8019  -0.4261  -0.0118   0.1903   3.3653
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.1355     3.0580  -3.968 7.23e-05 ***
## log_dis      -1.5266     0.4520  -3.377 0.000732 ***
## log_age       2.2585     0.6022   3.750 0.000177 ***
## log_rad       2.5736     0.4668   5.513 3.53e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 517.06  on 372  degrees of freedom
## Residual deviance: 225.57  on 369  degrees of freedom
## AIC: 233.57
##
## Number of Fisher Scoring iterations: 7
```

4 SELECT MODELS

	Sensitivity	Specificity	Precision	Recall	F1
Model.1	0.9111111	0.7708333	0.7884615	0.9111111	0.8453608
Model.2	0.9565217	0.9787234	0.9777778	0.9565217	0.9670330
Model.3	0.4693878	0.5000000	0.5111111	0.4693878	0.4893617
Model.4	0.8979592	0.7954545	0.8301887	0.8979592	0.8627451

Three models were explored in order to determine the best way to determine whether or not a neighborhood's crime rate was above or below the median crime rate. The most efficient model was the second model, with the first model being somewhat efficient, and the third model being least efficient.